

PATENT APPLICATION

METHOD AND APPARATUS
FOR STORAGE NETWORK MANAGEMENT

Inventors: Yuichi Taguchi, a citizen of Japan residing at
85 Rio Robles E., Unit 3411
San Jose, CA 95134

Naoki Watanabe, a citizen of Japan residing at
39-1, Takakura-machi, #102
Hachioji-shi
Tokyo, Japan

Assignee: HITACHI, LTD.
6, Kanda Surugadai 4-chome
Chiyoda-ku
Tokyo 101-8010, Japan
Incorporation: Japan

Entity: Large

METHOD AND APPARATUS FOR STORAGE NETWORK MANAGEMENT

CROSS-REFERENCES TO RELATED APPLICATIONS

- 5 **[01]** This application is related to commonly owned, co-pending U.S. Patent Application No. 09/841,710, entitled "Integrated Service Management System," filed April 24, 2001, and is incorporated herein in its entirety for all purposes.

BACKGROUND OF THE INVENTION

- 10 **[02]** The present invention relates to computer network systems and in particular to management of storage in a networking environment.
- [03]** The adoption of computer use in every aspect of our lives has created a great demand for storage capacity. Adequate data storage is an important component in most modern enterprises. The infrastructure technology that has developed to support the Internet has been
- 15 exploited to provide a variety of network-based storage solutions. Fig. 24 shows a range of storage architectures, including simple single storage/single server systems, and fibre channel technologies such as fibre channel (FC) storage area networks (SANs) and wider area FC SANs. Though not yet realized, the natural progression is toward a fully generalized IP-
- based (internet protocol) storage area network (shown in the figure by the dashed lines).
- 20 **[04]** Several types of storage devices are manufactured. They include RAID (redundant arrays of inexpensive disks) arrays, JBOD (just a bunch of disks) configurations which are spanning architectures that do not conform to the RAID definition, NAS (network attached storage) configurations, and so on. Accessing storage devices requires communication from the host, through the switching fabric, and through servers using any of a variety of protocols;
- 25 e.g. fibre channel (FC), iSCSI (internet SCSI - Small Computer System Interface), NFS, and so on.
- [05]** Given such a heterogeneous collection of intermediate components, it is not a trivial task to provision storage suitable for secured access, or with a guaranteed minimum throughput, and so on. As can be appreciated, each subsystem typically can be provided by a
- 30 different hardware and software manufacturers. Thus, provisioning suitable storage access typically requires detailed knowledge of each subsystem in order to provide an appropriate access path. For example, a different user interface ("agent") for configuring each

component typically is required. This necessitates the use of one or more operators to administer each component (e.g., host), and to collectively coordinate the configuration of the myriad components to connect the user's host system to her storage media with the access characteristics that she requires.

5 [06] It can be appreciated that there is a need to provide access to storage in a storage area network for users who have disparate storage needs in a storage system which subsystems have disparate administrative requirements. There is a need to facilitate the establishment and subsequent management of network paths in a storage area network based, for example, on security criteria.

SUMMARY OF THE INVENTION

10 [07] In accordance with the present invention, a storage path configuration can be established by maintaining a store of information representing the network configuration of a network. A storage service manager can establish a segment of the storage path
15 configuration with a host component of the network. A network agent can establish another segment of the storage path configuration with a switch component of the network. A storage agent can establish still another segment of the storage path configuration with a data storage component of the network. In accordance with an aspect of the present invention, the storage path configuration does not end at the network port of the storage facility, but rather
20 continues into the data storage component, reaching one or more physical devices (e.g., logical units) within the data storage component. In a particular embodiment of this aspect of the invention, the internal data path can be a path to a specific logical unit (volume) defined within the data storage component. Still another aspect of the invention is to specify a storage path configuration in abstract terms to represent characteristics of the access path, in
25 addition to the topological configuration of the physical data path.

BRIEF DESCRIPTION OF THE DRAWINGS

[08] Aspects, advantages and novel features of the present invention will become apparent from the following description of the invention presented in conjunction with the
30 accompanying drawings wherein:

Fig. 1 shows a storage service manager in relation to components comprising a storage network system in accordance with the present invention;

Fig. 2 is a schematic illustration of a hardware exemplar of the storage service manager of the present invention;

Fig. 3 shows an architectural block diagram representation of a software exemplar of the storage service manager of the present invention;

Fig. 4 shows an illustrative representation of the information comprising the policy information;

5 Fig. 5 shows an illustrative representation of the information comprising the storage pool information;

Fig. 5A illustrates a RAID example of a storage pool;

Fig. 6 shows an illustrative representation of the information comprising the storage path configuration information;

10 Fig. 7 shows an illustrative representation of the information comprising the storage service information;

Fig. 8 illustrates a configuration represented by the storage service information shown in Fig. 7;

15 Fig. 9 shows an illustrative representation of the information comprising the server configuration information;

Fig. 10 shows an illustrative representation of the information comprising the server service information;

Fig. 11 shows an illustrative representation of the information comprising the network topology information;

20 Fig. 12 shows an illustrative representation of the information comprising the network configuration information;

Fig. 13 shows an illustrative representation of the information comprising the network service information;

25 Fig. 14 is a high level flow diagram highlighting system status initialization in accordance with an illustrative embodiment of the present invention;

Fig. 15 is a high level flow diagram showing a policy definition process;

Fig. 16 is a high level flow diagram highlighting service request processing in accordance with an illustrative embodiment of the present invention;

30 Fig. 17 is a high level flow diagram highlighting storage path configuration processing in accordance with an illustrative embodiment of the present invention;

Fig. 17A is an illustrative example of a storage security enforcement rule;

Fig. 17B is an illustrative example of a port bandwidth calculation;

Fig. 17C is an illustrative example of an LDEV selection rule;

Fig. 18 is a high level flow diagram highlighting server configuration processing in accordance with an illustrative embodiment of the present invention;

Fig. 19 is a high level flow diagram highlighting network configuration processing in accordance with an illustrative embodiment of the present invention;

Fig. 20 shows an example of storage path configuration allocation at the system architecture level;

Fig. 20A shows an alternative set of abstract information for specifying a storage path configuration;

Fig. 21 illustrates a heterogeneous architecture adapted with a storage network management in accordance with an embodiment of the present invention;

Fig. 22 shows a configuration exemplar of switches comprising a IP network and an FC network; and

Fig. 23 shows a typical storage facility.

DESCRIPTION OF THE SPECIFIC EMBODIMENTS

[09] To provide a backdrop for discussing the embodiments of the present invention, refer for a moment to Fig. 23. The figure illustrates a typical storage area network (SAN) that can be adapted to operate in accordance with the invention. There are client components 2302 which can access various host components 2304. A typical function performed by host components is that of application servers. Application servers such as database servers tend to have intensive data storage requirements, and thus are suitable subjects for discussing the various aspects of the invention. However, it can be appreciated that other server (host) types 2304a can be adapted according to the invention. For example, an internet service provider may offer storage capacity for its registered users for hosting web sites, for uploading images, and so on. These systems also require large storage path configuration for its users. Thus, the general phrase “host component” will be understood to represent a variety of different server architectures and the like.

[10] A communication network 2312 typically provides a communication path between the clients 2302 and hosts 2304. Data storage components 2308 represent a myriad of storage device types and configurations from different models. The data storage components can be located in a room, they may occupy an entire building, they may be located in different cities or distributed across the globe. A communication component 2306, typically comprising various switches from different models, provides multiple data paths between the host components 2304 and the data storage components. All or a portion of the communication

component may be referred to as a switching fabric. The various illustrative embodiments of the present invention that will be discussed are components used to manage the network path between the hosts and data storage components through the switching fabric.

[11] Fig. 1 illustrates, in a generalized block diagram, a network architecture exemplar 100 according to the present invention. A storage service manager 102 serves as an entry point for an operator (administrator) to manage the various components in the network which comprise a “storage path configuration.” Through the storage service manager, an administrator can allocate to a user he requested “storage path configuration,” which in the context of the present invention, refers to the resources which can provide a specified level of I/O performance, capacity, operating characteristics, and so on by allocating appropriate components of the storage path configuration. This includes a server service, a network service, and a storage service. These components will be discussed in more detail below. The storage service manager 102 is typically a computer device configured with appropriate software components to provide the functionality that will be discussed. One such software component, for example, is a suitable user interface for accepting user input and displaying network configuration information.

[12] A communication backbone 112 connects the storage service manager 102 to various agent components. Server agents 124a and 124b interface with respective server (host) components 104 on behalf of requests from the storage service manager 102, in order to provide the “server services” called for by the storage service manager to satisfy a particular storage path configuration. Network agents 126a and 126b, likewise, interface with respective switches in response to requests from the storage service manager in order to provide the “network services” to satisfy the requested storage path configuration. One or more storage agents 128 interface with the various data storage components 108 comprising the network to provide a storage service appropriate for the requested storage path configuration.

[13] The number of agents for each component type (i.e., host, communication, and storage) will vary depending on the interfacing requirements for each component. For example, three switches from three different manufacturers or models may require three separate network agents, one agent specifically configured to interface to one of the switches. In a particular embodiment an agent can be implemented as device management software incorporated in the storage service manager 102. For example, it might be practical to use a conventional IP network agent and an FC network agent as a network agent, and to use vendor specific storage management tools as a storage agent. Alternatively, an agent

component can be a standalone computer, e.g., a PC suitably interfaced to its associated device (host, switch, data store) and running appropriate control software. Consequently, the block diagram elements for the agents illustrated in Fig. 1 represent some combination of software and/or hardware components. In a particular implementation, agent software components can be installed on the system components directly.

[14] Fig. 2 shows a generalized block diagram of the hardware architecture of the storage service manager 102. In a typical embodiment of this aspect of the invention, the network agent comprises a conventional computer system architecture. A processing device 202 (e.g., a CPU) provides the computing engine for performing the various tasks of the network agent. A common bus 212 provides a data path for data communication between the processing device and other components which comprise the network agent. The common bus can also include the cabling to provide power from a power supply 204 to the other devices. A network interface component 206 provides a suitable network connection to the host components 104, communication components 106, and data storage components 108 which comprise the network 100 to be managed.

[15] A mass storage component 222 serves as a data store to hold computer programs and various data needed for operation of the storage service manager 102 in accordance with the present invention. The functionality of the computer programs and the kind of data that is processed will be discussed in more detail below. A memory component 224 comprises some combination of random access memory (RAM), dynamic and static RAM, cache memory, non-volatile memory, flash memory, and so on. Input devices 226 might include user input devices such as a keyboard, mouse, etc. Output devices 228 might include monitors or the like, speakers for audio interfacing if appropriate, and so on.

[16] It can be appreciated that many other suitable configurations are possible. For example, the storage service manager 102 can have a distributed architecture. A part of the storage service manager may be a diskless workstation accessing the mass storage over a network. In this case, the common bus 212 can represent a network connection.

[17] In another embodiment, the processing device component 202 might represent a number of computing device components. For example, the processing device 202 might be a first computing device configured as a handheld device having some form of data connection (wired, wireless - radio, infrared, optical) to a second computing device configured as a server which is connected to the network 100 to provide remote access network management capability via the handheld device.

[18] Fig. 3 shows a high-level functional block diagram of the architecture of the storage service manager 102. A suitable interface provides for communication between the storage service manager 102 and the constituent components of the network architecture 100, namely, host components 104, communication components 106, and data storage components 108, to perform device configuration as requested by the administrator. A user interface module 302 provides access to the storage service manager 102 for an administrator (user) 10. This can be a graphical user interface (GUI). This is a common interface and may be appropriate in many situations. However, under other circumstances, a command line interface (CLI) may be more suitable. Still other interfaces are contemplated. For example, instead of a conventional human user, the “user” may be a machine. The interface can be a physically separate device, e.g., a handheld unit that communicates wirelessly with the storage service manager.

[19] A service configuration management module 316 receives input from the user interface 302 representative of the administrator’s commands. This module invokes the other constituent modules which comprise the storage service manager 102 that are needed to implement the administrator’s commands. A policy management module 314 provides access to a policy information data store 304.

[20] A server management module 342P, a network management module 362P, and a storage management module 382P, each contains the routines or procedures needed to control the configuration of the respective devices according to the administrator’s commands. For example, the server management module 342P can issue requests to a device (e.g., host bus adapter) on a host machine to configure the device. Similarly, the network management module 362P can interact with the various switching components that comprise the network to implement the administrator configuration commands. The storage management module 382P interacts with the various storage components to provide the storage capacity and characteristics required by the administrator.

[21] A communication interface is provided for each of the agents. These interfaces 342I, 362I, and 382I represent any interface (e.g., fiber channel, SCSI, ATA, and so on) that is suitable for a given device. Typically, communication occurs via standard network protocols such as TCP/IP, HTTP, LDAP and so on. These functional modules can be implemented using any conventional combination of hardware and software. In any given particular implementation, these modules can comprise one or more components. For example, different components may be needed for communicating with equipment from different

manufacturers. Custom components may be needed, for example, if a vendor provides a proprietary configuration port.

[22] Fig. 4 shows the data comprising policy information stored in the policy information data store 304 of Fig. 3. This information is defined by the administrator, and it constitutes the range of capacity and performance characteristics of different storage path configurations that are deemed appropriate for the storage facility that is being managed. For example, a commercial enterprise will likely implement policies for its internal storage needs that are different from the policies of a storage service provider. It can be appreciated therefore, that the information described herein may vary from one particular implementation to another. Moreover, the data structures shown in this figure and in the subsequent figures are a logical representation of the relevant information. Those of ordinary skill in the relevant art will appreciate that numerous implementation-specific designs for storing and accessing the data are possible.

[23] A service policy information table 402 describes the various storage path configuration policy levels that are defined for the system. Each service policy level is identified by a policy ID. These policies are defined by the administrator and represent the storage access policies of the storage service facility. Every user who desires a particular storage path configuration will be allocated the appropriate resources according to the service policy desired by the user. The service policies typically are pre-defined policies. For example, a storage server provider may define various levels of storage path configuration and “sell” those storage path configurations to users. New policies can be defined if a desired storage path configuration is not already defined.

[24] A service policy comprises three sub-policies: a storage sub-policy, a network sub-policy, and a server sub-policy. According to the present invention, each service policy is associated with a set of storage sub-policies, a set of network sub-policies, and a set of server sub-policies. Thus, as can be seen in Fig. 4, the service policy information table 402 includes a “pointer” field or some other indexing information to each of respective sub-policy tables, 404, 406, and 408. Having multiple sub-policies allows the administrator to perform tradeoffs among storage, network, and server resources for a given service policy. For example, for a given service policy, a first user might want higher capacity storage, but is willing to sacrifice network speed. A second user asking for the same service policy may be satisfied with lower capacity storage, but want higher network speed.

[25] A storage path configuration policy table 404 defines different storage sub-policies. A policy ID field contains unique identifiers for each storage sub-policy. A storage security

field indicates whether LUN (logical unit) masking is enabled (ON) or not (OFF). In a particular embodiment of the invention, the storage devices use the RAID (redundant array of inexpensive disks) architecture. A RAID level field serves to define the RAID level for a particular storage sub-policy. A cache field specifies the size (typically in MBytes) of storage space for the particular storage sub-policy. A port priority field is a ranking parameter which specifies the priority to prioritize the specific network transferred on the network port. A disk speed (units of RPM) represents the speed of the disk drive corresponding to a particular storage sub-policy. A storage pool field is a parameter which specifies a disk group that is pooled for a specific purpose. For example, a “financial” pool can be created for the subsequent use of a financial database application. The storage pool field is a pointer or other suitable index into a storage pool information table 384a discussed below in Fig. 5.

[26] A network configuration policy table 406 defines the network sub-policies. Each network sub-policy is uniquely identified by a policy identifier stored in a policy ID field. A network security field specifies whether a network security function (e.g., WWN zoning or VLAN) is being applied (ON) or not (OFF). A bandwidth field specifies the total bandwidth applied to the network path. A “num of path” field specifies the number of network paths for accessing the storage.

[27] The server sub-policies are stored in a server configuration policy table 408. Each sub-policy is identified by a unique policy identifier stored in a policy field. Computer resources are specified by a CPU field, a memory field, and a processor field. The CPU field can specify the percentage of CPU utilization allowed for a given policy. Thus, a user can be given up to 60% of the CPU time for sub-policy “Policy0001.” The memory field, likewise, specifies the memory utilization that is permitted for a given sub-policy. The processors field specifies the number of parallel processors (if available) to be used for a particular application. A platform field specifies a particular server platform. The information in this field determines the storage disk format.

[28] Fig. 5 shows the data comprising information stored in the storage pool information data store 384a of Fig. 3. In a particular embodiment of the invention, storage resources can be managed in groups that are determined by its usage. For example, a “financial pool” can be created for the use of financial database application service. A “logical” storage pool comprises of a set of LDEVs (logical devices). An LDEV is also a “logical” part of “physical” disk group. The disk group comprises a set of hard disk drives.

[29] Referring for a moment to Fig. 5A, an implementation of a RAID volume is shown to illustrate this aspect of the invention. The figures shows the implementation of a storage pool 512 using RAID arrays 516 and 518. The physical disks of the RAID arrays define a set of logical devices 514 which collectively define the storage pool. A group of disks 516a in

5 RAID array 516 defines a logical device 514a.

[30] Returning now to Fig. 5, a storage pool information table 502 defines the storage pools that are available for allocation, according to a specific storage access policy. A pool ID field stores unique identifiers for the different storage pools. A total capacity field indicates the storage capacity for a given storage pool. A free capacity field indicates the
10 amount of free space in the storage pool. An LDEV field points or otherwise identifies a logical device information table 504. This table defines the logical devices that comprise the storage pool.

[31] The logical device information table 504 includes an ID field to uniquely identify each logical device (LDEV) which constitutes a storage pool. A capacity field specifies the
15 total storage capacity of each logical device. A consumed field is a boolean type of parameter which specifies YES (TRUE, etc.) if the LDEV is allocated to the host. The free capacity value can be computed by subtracting the capacity of all of the allocated (consumed) LDEVs from the total capacity. A disk group field stores the identifiers of the physical disk groups (e.g., in a RAID array) that comprise the logical device.

[32] A disk group information table 506 describes each of the physical disk groups. A disk group ID field contains an identifier for each disk group. A RAID level field indicates the RAID level at which the physical disk group is operating. A capacity field indicates the total storage capacity of an LDEV. A “num of LDEVs” field indicates the number of logical devices defined by the physical disk group. A disk speed (units of RPM) specifies the
25 rotational speed of the physical disks in the physical disk group. A format field specifies the formatting of a disk; for example, the disk can be formatted for NTFS in the case of UNIX-based OSs, or S/390 in the case of mainframes.

[33] Fig. 6 shows the data comprising information stored in the storage path configuration information data store 384b of Fig. 3. The use of storage can be determined based on the
30 network port information to be assigned, the allocated cache memory size, and the volume information contained in the storage path configuration table 602. Each storage device can be identified by information in the storage ID field. The ports field is a suitable pointer to a network ports sub-table 604. This sub-table identifies the ports that are assigned to the storage device. A cache field specifies the amount of cache memory used in the storage

device for buffering the data being transferred to and from the storage device. A volume field is a pointer to a volume information sub-table 606. Each entry contains one or more pointers to an LDEV table 608, to identify the one or more LDEVs which comprise a LUN.

[34] The network ports sub-table 604 comprises the port IDs that are assigned to a storage

5 service. This table includes a bandwidth capacity field that indicates the total bandwidth capacity for each port. An assigned bandwidth field indicates the total current bandwidth that has been assigned (allocated) to the port. Each storage path configuration that is assigned to a port is also allocated a certain bandwidth on the assigned port. This “assigned bandwidth” field shows the total bandwidth for a port that has been allocated to all storage path

10 configurations using that port. It is noted that there are situations where multiple ports can be assigned to a single storage volume. Also, a single network port can be shared by multiple storage volumes.

[35] The volume information sub-table 606 comprising data storage information. A volume can be created with either a LUN or a list of LDEVs. In case of the OPEN system, a
15 volume corresponds to a LUN that in turn comprises a set of LDEVs. In that case, the table entry points to the list of constituent LDEVs. For example, the volume LUN00A0 comprises two LDEVs. Consequently, a suitable pointer data mechanism is provided (e.g., linked list, or the like) point to the constituent LDEVs. It is noted that the capacity size of a LUN can be calculated as the sum of the size of LDEVs.

20 [36] Fig. 7 shows the data comprising information stored in the storage service information data store 386 of Fig. 3. As mentioned above, a storage service is one of three components which comprise a storage path configuration; the other two components are a server service and a network service, which will be discussed below. A storage service is a set of reserved storage-related resources that can be allocated to a requested storage path configuration. A
25 storage device table 702 identifies storage services defined by the administrator. Each storage service is identified by an identifier contained in a service ID field. A storage ID field identifies the storage device in the storage path configuration information data store 602. A cache field indicates the total cache memory size reserved for the given storage service. A pointer structure is provided for each storage service to point to one or more entries in a
30 storage port table 704. This identifies the port(s) associated with a given storage service.

[37] The storage ports table 704 includes, a storage world-wide name (WWN) port field which stores the WWN designation of the port. The WWN is an IEEE specification that provides world wide, unique identification of fabric devices. A port ID field identifies an entry in the network ports sub-table 604 to provide information for each port associated with

a storage service. A bandwidth field indicates the assigned bandwidth for a given storage service. Each port is associated with a LUN. Thus, each entry in the storage port table 704 includes a pointer to a LUN table 706 that identifies a LUN. Each entry in the LUN table includes a LUN identifier in the volume information sub-table 606. Each LUN is associated with one or more entries in a host port WWN table 708. The host port WWN lists the WWN designation of each HBA on the host.

[38] Fig. 8 shows the storage services defined in Fig. 7. For example, the storage service Srvc0001 provides access to the storage device Strg00A0, as shown in the storage device table 702. Tracing through the storage port table 704 and the LUN table 706 to the host port WWN table, it can be seen in Fig. 8 that storage service Srvc0001 provides access to storage device Strg00A0 via three host ports, namely, WWN0A00, WWN0A01 and WWN0A02. The storage service also provides access via the storage ports WWN00A0 and WWN00A1. It is noted, that storage service in accordance with the present invention provides access to specific LUNs defined in the storage device. As the LUN table 706 indicates, and as shown in Fig. 8, the storage service Srvc0001 provides access to LUN00A0 and LUN00A1. The present invention allows the administrator to reach a LUN from a server host seamlessly without individual device management operation.

[39] Fig. 9 shows the data comprising information stored in the server configuration information data store 344 of Fig. 3. A server host computer configures its resource consumption of network bandwidth and processor power. Within a fiber channel network, for example, a server computer is equipped with one or more host bus adapters (HBAs). An HBA is a network interface for fiber channel platform. As indicated above, each HBA is associated with a World Wide Name (WWN) designation as an identifier. Each server is identified by a server ID and stored in the server ID field of a server table 902. A platform field indicates the underlying operating system. A ports field points to a host port table 904 which lists each host port for the given server. Each host port in the host port table is associated with a WWN designation in the host port field. A bandwidth field in the host port table specifies the bandwidth of the host port. A memory field in the server table 902 specifies the amount of memory contained in the server. A processors field points to a processor table 906 that lists identifiers of the processors which comprise the given server.

[40] Fig. 10 shows the data comprising information stored in the server service information data store 346 of Fig. 3. Recall, that a storage path configuration comprises three components: a storage service (discussed above), a server service, and a network service (which will be discussed below). The server service represents the component of a storage

path configuration relating to the computing resources of a server host. A server service can be provided either by a single server host computer or by multiple server host computers.

Fig. 10 illustrates a case where one server (Srvr0A00) supports two services. However, there can be a case that multiple servers (Srvr0A00 and Srvr0B00) are assigned to provide a single service (Srvc0001). This illustrates an aspect of the present invention wherein logical computer resources can be allocated to several server services.

[41] A server service table 1002 maintains the server services that have been defined by the administrator and which can be allocated as a components of storage path configurations. A service ID field identifies each server service. A server ID field identifies each server in the network, and relates to the server table 902. A memory field indicates the amount of memory (MB) reserved to a given server service. A CPU share field indicates the CPU processing capacity (as a percentage of the total CPU capacity) allocated to a given server service. A num of processors field indicates the number of processors allocated to a given server service. An assigned processors field points to a processor table 1004 that identifies a list of processor IDs of processors allocated to a given server service. A num of ports field indicates the number of ports assigned to a given server service. An assigned ports field is a pointer to a list 1006 of host port IDs. As noted above, network ports can be identified by a WWN designation on a fiber channel network.

[42] Fig. 11 shows the data comprising information stored in the network topology information data store 364a of Fig. 3. A network topology map can be represented by a combination of a source object 1112 having a node ID 1102 and a port ID 1104, and a destination object 1114 also having a node ID 1106 and a port ID 1108. A “node” refers to a particular component object such as a host 102, a switch 104, or a data storage component 108. It is noted that a host can be logically comprised of two or more nodes, each having its own ID. Similarly, a switch can be logically viewed as having two or more nodes, and a similar possibility exists for the data storage components. The node ID can be some arbitrarily assigned identifier.

[43] The port ID can be an arbitrarily assigned identifier. Alternatively, the port ID can be a World Wide Name in the case of a fibre channel port to which the node is connected. In an IP environment, the port ID can include the MAC address or a combination of an IP address and an IP subnet mask. It can be appreciated that a node can have more than one port.

[44] Hosts 102 and switches 106 can be categorized as source objects 1112. Data store 108 and again switches can be characterized as destination objects 1114. This characterization is determined from the point of view of data flowing downstream from host

to data store. The data flow is merely an adopted convention. A reverse view can be adopted as well, where the data store is a starting point and the host is the end point. Either view, can serve to represent the topology of the network. It can be appreciated that still other conventions can be employed to represent the network topology.

5 [45] Fig. 12 shows the data comprising information stored in the network configuration information data store 364b of Fig. 3. A network devices table 1202 identifies each network by an identifier in a network ID field. A type field indicates the network platform. A security field indicates the type of security function provided by a given network. A device objects field lists the devices comprising the network in a devices table 1204, each device
10 being identified in an object ID field.

[46] Fig. 13 shows the data comprising information stored in the network services information data store 364b of Fig. 3. As mentioned above, a network service is a component of a storage path configuration, the other components being a storage service and a server service, both of which have been discussed above. The network service consists of one or
15 more network paths that connect a server host and a storage device. A network table 1302 identifies each network service in a service ID field. A path ID identifies, for a given network service, a network path between a port in the server (host) and a port on a storage device. A host port field identifies the host port to which a given path is “connected,” or otherwise in data communication. A storage port field identifies the storage port to which a
20 given path is “connected” or otherwise in data communication. A zone ID field is provided in the case of a fibre channel network path, and identifies different zones in a multi-zone configuration. It can be appreciated of course that other security implementations on other network platforms can be accommodated. A bandwidth field indicates the bandwidth for a given path. An objects field is a pointer to an objects table 1304, which identifies the
25 network objects that constitute a given path.

[47] The objects table 1304 includes a switch ID field which identifies a constituent switch object for the given path. An inport ID field and outport ID field identify the respective in and out ports to which the given path is connected. Conventionally, the inport and outport are defined relative to a direction from the storage device port to the host (server) port. The
30 tables shown in Fig. 13 represent the configuration shown in Fig. 8.

[48] Fig. 14 is a high level flow diagram highlighting the main process steps for initializing device configuration information in the storage service manager 102. When the storage service manager boots up or otherwise begins processing, in a step 1402, it first has to discover its universe of network objects (servers, switch elements, storage components) in

order to initialize the device configuration status. Referring for a moment to Fig. 1, the storage service manager 102 is shown accessing and/or controlling network objects through the agents, namely, the server agents 124, the network agents 126, and the storage agents. However, it can be appreciated, in another implementation, that the storage service manager
5 102 can be configured to communicate with the network objects directly via vendor dependent communication methods; e.g., switch vendors typically provide their own administration software to access their equipment, similarly for large storage components. It can be appreciated for those cases, that the storage service manager 102 can be configured to interact with the vendor software, via a machine interface (if available), or simply by
10 emulating a human user.

[49] At a step 1404, the storage service manager 102 sends request messages to the agents to discover the network objects. Alternatively, where vendor access software is provided, the storage service manager can interact with the software to obtain the information. At steps 1432 and 1434, each agent (or the vendor provided software) receives a message that request
15 device information. Each agent or vendor software retrieves the information and returns one or a series of messages that contains the requested device configuration status information, step 1436. The communication can use standard protocols, such as SNMP, HTTP, and so on; CIM (Common Information Model) is a good example for transferring message data in the standard data model. From steps 1408 to 1416, the storage service manager 102 updates the
20 various local information repositories according to the device configuration information received at step 1406. As an observation, if at step 1406, the storage pool is not defined by the storage agent, the storage service manager 102 can allocate storage resources to define one or more storage pools as needed.

[50] Fig. 15 shows the basic steps for defining, modifying, or otherwise managing the
25 policy information. This is a task for the administrator of the storage service management system. A suitable presentation of the policy information contained in the policy information data store 304 is made to the administrator. Typically, this is a GUI based presentation. The administrator can then navigate through the different policies, create new policies, delete policies, modify policies, and so on. These changes are then reflected back to the policy
30 information data store. For example, the administrator may define one or more policies for financial applications.

[51] Fig. 16 shows a high level process flow for handling a storage path configuration setup request. When a user has a need for storage, she will make a request to the administrator for a storage path configuration. The administrator will then access the storage

service manager 102 to allocate the resources (e.g., server component, network component, and storage component) to provide the storage path configuration being requested. A display is presented, in a step 1602, to the administrator in a suitable interface. In a step 1622, the administrator selects a service configuration policy according to the requested level of service.

5 For example, the administrator may choose a “financial application service” policy for the purpose of creating a service for financial use. As another example, a “Gold service” policy might be defined for customers who are willing to pay a premium for high speed and/or high capacity storage access.

[52] In a step 1604, the policy management program 314 on the storage service manager

10 102 loads the policy information (Fig. 4) that is requested by an administrator. At a step 1624, the administrator specifies a storage capacity size to be allocated from the storage pools.

A set of storage pool identifiers which identify the storage devices which satisfy the requested capacity are determined and stored in memory, in step a 1606. At a step 1626, the administrator selects one or more servers from which the user can access her storage. The

15 server IDs of the selected servers are stored in memory, in a step 1608. Then, in a step 1610, the storage service manager 102 starts a process to allocate resources as requested. This includes initiating processing in the storage management module 382P, the server management module 342P, and the network management module 362P to allocate the necessary components of the requested storage path configuration. Processing in each of
20 these modules is discussed respectively in connection with Figs. 17, 18, and 19.

[53] Recall that a storage path configuration comprises three components: a storage service, a server service, and a network service. Fig. 17 is a generalized flow diagram highlighting the major steps for allocating a storage service component for a requested storage path

25 configuration. At steps 1702 and 1704, the storage management module 382P looks up a policy enforcement rule to calculate actual parameter values to enforce on the storage device.

The storage path configuration policy 404 shown in Fig. 4 includes some parameters that are defined by an administrator. For example, if security is ON, the storage management module has to decide a security type to apply. Thus, an aspect of the invention is the use of “rules” as the basis for identifying performance requirements and security requirements for a given

30 policy. By comparison, a pure policy-based determinations involve identifying a set of policies, and simply associating desired performance and security criteria for each policy.

Rule-based determination involves a more heuristic approach, where decisions are based on execution of the rule to evaluate the logic and/or computations comprising the rule.

[54] Fig. 17A shows a typical rule that can be applied to a security function that might be implemented on a LUN. For the rule exemplar shown in Fig. 17A, the security function is provided in a line of storage systems (Freedom Storage series) manufactured and sold by the Assignee of the present invention and is referred to as “LUN masking.” The illustrated rule
5 limits storage accesses only from permitted host WWNs. Referring to Fig. 4, the “storage security” field in the storage configuration policy table 404 can be tested as a flag in an “enforcement rule” using a logic statement.

[55] Fig. 17B shows a code fragment for converting a rank parameter into an actual bandwidth parameter. This figure illustrates that a rule can be a computation. In this
10 particular implementation of the invention, port priority is set by its rank. This facilitates the administration function. However, port priority may have to be assigned based on a maximum bandwidth (MB/sec); for example, the “access priority” function used in the Freedom Storage series sets a maximum bandwidth on a network port. This function can limit the maximum bandwidth that can be consumed by traffic from one network port on the
15 storage to one host WWN. The use of a rule in this situation allows a translation of one parameter (“port priority”) to another parameter (“bandwidth”) which can be used to select a port based on bandwidth.

[56] After an administrator inputs a policy and a capacity size, the storage service manager
102 can automatically determine the storage LDEVs to assign. For example, suppose “disk speed” is set to ‘10000’ in the policy and a ‘20000’ Mb capacity is requested for storage pool ‘Pool0010’. The storage service manager refers the storage pool information tables in Fig. 5 and checks the ‘free capacity’ field for the storage pool identified as ‘Pool0010’. It finds that the disk group DGRP1053 has ‘10,000’ rpm drives. It searches the logical device
20 information table 504 for free LDEVs. This is accomplished by checking the consumed field, which indicates whether the LDEV has been assigned to a storage path configuration. An LDEV selection rule, such as the rule illustrated by the code fragment of Fig. 17C, can be used.

[57] At a step 1706, the storage service manager 102 sends a request to a storage agent 128
30 to create a new volume or to expand an existing volume. The request message contains a storage ID, LDEV list, and a port ID to assign.

[58] At step 1722, the storage agent 128 for the storage device identified by the storage ID creates a new volume as requested. The storage agent returns the storage port WWN.

[59] At a step 1708, the storage service manager 102 makes a requests to apply a security measure according to a security policy if there is one associated with the requested storage

path configuration. Referring again to Fig. 4, the service policy for a given storage path configuration may be associated with a security policy for storage devices, or the user may explicitly request a particular security policy for her storage path configuration. This request is sent to the storage agent to implement the requested security measure. Thus, at a step 1724, the storage agent 128 applies the requested storage security. For example, in Freedom Storage, a storage security function can restrict storage access to permit access only from the host port WWN.

[60] At a step 1710, the storage service manager 102 can make a request for a port access priority control. The request is sent to the storage agent 128, wherein at a step 1724, the storage agent applies an access bandwidth on the network port that reserves bandwidth for traffic from a host WWN to the storage LUN.

[61] At a step 1712, the storage service manager 102 requests a cache memory size for the requested storage path configuration. The request is sent to the storage agent which, in a step 1728, configures a cache of appropriate size in the storage device(s). At a step 1730, configuration status information can be sent to the storage service manager 102 to confirm all of the settings. Finally, at a step 1714, the storage service manager 102 updates the storage path configuration information data store 384b, the storage service information data store 386, and the storage pool information data store 384a with the information obtained above.

[62] Fig. 18 is a generalized flow diagram highlighting the major steps for allocating a server service component for the requested storage path configuration. In a step 1802, the storage service manager 102 calculates server computing resources that are defined by the policy. The CPU share and processor IDs are determined as satisfying the service level. Network bandwidth may be guaranteed if the port priority on the storage port is clear. The storage service manager 102 chooses a network port on the server that has the necessary bandwidth capacity to satisfy the bandwidth guaranteed on the storage. Consider the network example of Fig 8, in which the storage ports have a 2,000 Mb/sec bandwidth. In this case, at least two 1,000 Mb/sec network ports on the server must be assigned to provide a 2,000 Mb/sec bandwidth.

[63] At a step 1804, the various objects in the server are selected, including desired processors, network ports on the servers (e.g., HBA ports), and memory size. At a step 1806, the storage service manager 102 sends a request to a server agent 124 to allocate suitable host ports. The request message contains a server ID and a processor list.

[64] At step 1822, the server agent 124 identifies the requested processor resources as being allocated. At a step 1808, the storage service manager 102 makes a request to allocate

the network ports on the servers. This request is sent to the server agent 124, at a step 1824, which marks the specified network ports as being allocated.

[65] At a step 1810, the storage service manager 102 can make a request for a desired memory size. This specifies the amount of memory to be allocated to handle the requested storage path configuration. The request is sent to the server agent 124, wherein at a step 1826, the memory allocation is noted.

[66] At a step 1812, the storage service manager 102 updates the server configuration information data store 344 and the server service information data store 386 with the information obtained above.

[67] Fig. 19 is a generalized flow diagram highlighting the major steps for allocating a network service component for the requested storage path configuration. In a step 1902, the storage service manager 102 calculates the configuration parameters in accordance with the policy associated with the storage path configuration; for example, the number of paths to create. At a step 1904, the storage service manager calculates a network route, comprising a list of network objects, from the server to the storage.

[68] At a step 1906, the storage service manager 102 sends a request to one or more network agents 126 to set the path on the network, including any requested security capability. Referring again to Fig. 4, the service policy for a given storage path configuration may be associated with a security policy for the network path, or the user may explicitly request a particular security policy for her storage path configuration. The request message contains a switch ID (a switch to set the path), an inport ID (an ingress port on the switch), an output ID (an egress port on the switch), and a zone ID if security is desired. At step 1922, the network agent(s) 126 sets the path on the switch as requested in step 1904. At a step 1908, the storage service manager 102 makes a bandwidth reservation request using the path ID determined in step 1906 to allocate the desired bandwidth for the path. This request is sent to the network agent(s) 126, which at a step 1924, controls the switch to guarantee the bandwidth on the path as requested.

[69] At a step 1910, the storage service manager 102 updates the network configuration information data store 364b, the network service information data store 366 with the information obtained above.

[70] Fig. 20 is a high level illustrative example of a storage path configuration scenario. In response to a user request for storage service, an administrator will enter her administration input 2002 to setup the requested storage path configuration (Fig. 17). An aspect of the present invention is that the administrator can specify a storage path configuration in terms of

abstract characteristics of the desired configuration, such as capacity and access policy shown in Fig. 20. Fig. 20A shows an alternative set of inputs to emphasize that other kinds of abstract characterizations can be provided by an administrator. The illustrative example of Fig. 20A shows the administrator can specify characteristics such as a RAID level, a security parameter, an application (database application, for example), and so on.

[71] Returning to Fig. 20, the selected service policy in this example is for “Financial DB Service.” This abstract characterization, as can be seen in the figure, represents a pre-defined configuration 2004. Based on the selected service policy, the server agent 124 allocates a suitable server service 2024a based on parameters 2014 of the server service component of the storage path configuration. Similarly, the network agent 126 allocates a suitable network service component 2026 based on parameters 2016 of the network service component. The network service component parameters are based on the selected service policy 2004. The storage service component 2028 is allocated by the storage agent 128 based on parameters 2108 of the storage service component, which in turn are based on the selected service policy. Recall, that an aspect of the present invention is the allocation of an internal data path within the data storage components 108 (Fig. 1) to a specific physical device, or devices. For example, the data path may lead to a logical unit comprising a plurality of physical devices that are logically organized as a single volume.

[72] Fig. 21 illustrates the adaptability of the storage service manager 102 of the present invention. The figure illustrates a heterogeneous operating environment. At the host end, the storage service manager may be faced with a plurality of server architectures. The configuration exemplar shown in the figure illustrates a Microsoft Win server 2104a and a Sun Microsystems Solaris server 2104b. Generally, different host architectures may provide widely varying capabilities and very different interfaces for system configuration, even for common administrative functions. Consequently, the server agent (e.g., 124a, Fig. 1) component may comprise a customized server agent for each such host, as shown in the figure (and also in Fig. 1). For example, one such server agent might be a Win server manager 2124a that facilitates volume mounting and volume size expansion. Another server agent might be a Solaris server manager 2124b to provide similar capability and other configuration options.

[73] Turning to Fig. 22 for a moment, it can be seen that host servers and storage devices can be located throughout the network. The arrangement need not be purely hierarchical as shown in the configuration of Fig. 21. In Fig. 22, it can be seen that host servers can be connected to each switch 2206a, 2206b. Different storage devices can be connected to each

switch. For example, an iSCSI storage device and a NAS storage facility can be accessed over the IP network 2206a. A RAID storage system and a JBOD storage path configuration can be accessed over the FC network 2206b. Any of the hosts can access any of the storage units. The storage service manager 102 (Fig. 1) of the present invention can allocate storage path configuration using the configuration information stored in the various repositories shown in Fig. 3. The method and apparatus to maintain a consistent path through IP network and fiber channel network is discussed in more detail in U.S. Patent Application No. 09/841,710.

[74] Fig. 21 further shows that the switching fabric can comprise plural switch architectures. As shown the figure, switches are provided for an IP network 2106a and an FC network 2106b. Again, each switch architecture may have its own network agent to interface with the respective switch. For example, a switch in an IP network agent 2126a may provide virtual LAN capability that can configured and maintained by the corresponding network agent. Fabric zoning in a switch comprising the FC network can be exploited to facilitate implementation of security policies by a FC manager 2126b. In fact, it is possible that different manufactures of the same switch type will have different interfaces for administration. The network agent component 126 (Fig. 1) in that case may comprise a separate network agent component to properly interface with each switch type.

[75] Similarly, a storage agent can be provided for each storage component. For example, Fig. 21 shows an iSCSI storage device 2108a and a RAID array 2108b. Each storage architecture has its respective storage agent (2128a, 2128b) to access the storage path configuration capabilities of each architecture. For example, a RAID manager 2128b can support volume creation, path mapping, and LUN masking.